CLAIMS

1. A test system for testing an in-test host's support of USB peripherals, the test system comprising:

one or more USB interfaces configured to communicate with one or more USB ports of the in-test host to communicate USB messages with the in-test host;

a network interface configured to communicate with a peripheral emulator using a network communications protocol;

operating logic configured to perform actions comprising:

receiving USB command messages from the in-test host;

sending the received USB command messages to the peripheral emulator through the network interface using the network communications protocol; and

receiving USB response messages from the peripheral emulator through the network interface using the network communications protocol;

sending the received USB response messages through the one or more USB interfaces to the in-test host.

2. A test system as recited in claim 1, further comprising the peripheral emulator, wherein the peripheral emulator is programmed to emulate one or more USB peripherals.

3. A test system as recited in claim 1, further comprising the peripheral emulator, wherein the peripheral emulator is programmed to emulate HID, bulk, and isochronous USB peripherals.

4. A test system as recited in claim 1, further comprising the peripheral emulator, wherein the peripheral emulator comprises a general-purpose computer programmed to emulate one or more USB peripherals.

5. A test system as recited in claim 1, further comprising the peripheral emulator, wherein the peripheral emulator comprises a general-purpose computer programmed to emulate HID, bulk, and isochronous USB peripherals.

6. A test system as recited in claim 1, further comprising the peripheral emulator, wherein:

the peripheral emulator comprises a general-purpose computer;

the general-purpose computer is programmed to emulate one or more USB peripherals; and

the general-purpose computer is further programmed to generate USB response messages that test the in-test host with ranges of USB peripheral parameters.

7. A test system as recited in claim 1, further comprising the peripheral emulator, wherein:

the peripheral emulator comprises a general-purpose computer;

the general-purpose computer is programmed to emulate one or more USB peripherals; and

the general-purpose computer is further programmed to generate abnormal USB response messages in order to test the in-test host with such abnormal USB response messages.

8. A test system as recited in claim 1, wherein:

a particular USB command message is designated for a particular one of a plurality of different emulated peripheral devices;

the network communications protocol supports a plurality of logical ports;

the operating logic maintains a correspondence between emulated peripheral devices and logical ports; and

the operating logic sends said particular USB command message to one of the logical ports that corresponds to said particular one of the plurality of different emulated peripheral devices.

9. A test system as recited in claim 1, wherein the one or more USB interfaces comprise at least four USB interfaces.

10. A test system as recited in claim 1, wherein the USB messages comprise HID, bulk, and isochronous USB messages.

11. A test system as recited in claim 1, wherein the network interface comprises an Ethernet interface.

12. A test system as recited in claim 1, wherein the network communications protocol comprises an Ethernet communications protocol.

13. A test system as recited in claim 1, wherein the network communications protocol comprises an IP protocol.

14. A test system as recited in claim 1, wherein the network communications protocol comprises UDP over IP.

15. A test system for testing an in-test host's support of peripheral devices that utilize a peripheral communications protocol, the test system comprising:

a peripheral interface configured to communicate with a peripheral device port of the in-test host using the peripheral communications protocol;

a network interface configured to communicate with a peripheral emulator using a network communications protocol;

operating logic configured to perform actions comprising:

receiving outgoing peripheral device messages through the peripheral interface from the in-test host using the peripheral communications protocol;

sending the received outgoing peripheral device messages through the network interface to the peripheral emulator using the network communications protocol;

receiving incoming peripheral device messages through the network interface from the peripheral emulator using the network communications protocol; and

sending the received incoming peripheral device messages through the peripheral interface to the in-test host using the peripheral communications protocol.

16. A test system as recited in claim 15, further comprising the peripheral emulator, wherein the peripheral emulator is programmed to emulate one or more USB peripherals.

17. A test system as recited in claim 15, further comprising the peripheral emulator, wherein the peripheral emulator comprises a general-purpose computer programmed to emulate one or more USB peripherals.

18. A test system as recited in claim 15, further comprising the peripheral emulator, wherein the peripheral emulator comprises a general-purpose computer programmed to emulate one or more USB peripherals and to generate peripheral device messages that test the response of the in-test host to ranges of USB peripheral parameters.

19. A test system as recited in claim 15, further comprising the peripheral emulator, wherein the peripheral emulator comprises a general-purpose computer programmed to emulate one or more USB peripherals and to generate peripheral device messages that test the response of the in-test host to abnormal conditions.

20. A test system as recited in claim 15, wherein:

a particular outgoing peripheral device message is designated for a particular one of a plurality of different emulated peripheral devices;

the network communications protocol supports a plurality of logical ports;

the operating logic sends said particular outgoing peripheral device message to one of the logical ports that corresponds to said particular one of the plurality of different emulated peripheral devices.

21. A test system as recited in claim 15, wherein the peripheral interface comprises a USB interface.

22. A test system as recited in claim 15, wherein the received outgoing peripheral device messages comprise USB messages and the received incoming peripheral device messages are sent to the in-test host as USB messages.

23. A test system as recited in claim 15, wherein the peripheral communications protocol conforms to the USB standard.

24. A test system as recited in claim 15, wherein the network interface comprises an Ethernet interface.

25. A test system as recited in claim 15, wherein the network communications protocol comprises an Ethernet communications protocol.

26. A test system as recited in claim 15, wherein the network communications protocol comprises an IP protocol.

27. A test system as recited in claim 15, wherein the network communications protocol comprises UDP over IP.

28. A method of testing an in-test host's support of USB peripherals, comprising:

receiving USB command messages from the in-test host;

packaging the received USB command messages in command data packets formatted in accordance with a network communications protocol;

sending the command data packets to one or more peripheral emulators over network communications media;

receiving response data packets from the one or more peripheral emulators over the network communications media, wherein the response data packets are formatted in accordance with a network communications protocol;

unpackaging USB response messages from the received response data packets;

sending the unpackaged, USB response messages to the in-test host.

29. A method as recited in claim 28, further comprising emulating one or more different USB peripherals within the one or more peripheral emulators to create the incoming USB messages.

30. A method as recited in claim 28, further comprising creating abnormal USB response messages in response to the packaged USB command messages and packaging said abnormal USB response messages in the response data packets in order to test the in-test host's ability to handle such abnormal USB response messages.

31. A method as recited in claim 28, wherein the network communications protocol comprises an Ethernet communications protocol.

32. A method as recited in claim 28, wherein the network communications protocol comprises an IP protocol.

33. A method as recited in claim 28, wherein the network communications protocol comprises UDP over IP.

34. A computer comprising:

a network interface configured to receive network communications formatted in accordance with a network communications protocol;

one or more processors programmed to perform actions comprising:

emulating one or more USB peripherals in order to create USB response messages;

packaging the USB response messages in response data packets formatted in accordance with the network communications protocol; and

sending the response data packets over network communications media.

35. A computer as recited in claim 34, wherein the one or more processors are programmed to perform further actions comprising:

receiving data packets over the network communications media, wherein the data packets are formatted in accordance with a non-USB network communications protocol; and

unpackaging the USB response messages from the data packets.

36. A computer as recited in claim 34, wherein the one or more processors are programmed to emulate HID, bulk, and isochronous USB peripherals.

37. A computer as recited in claim 34, wherein the network communications protocol comprises an Ethernet communications protocol.

38. A computer as recited in claim 34, wherein the network communications protocol comprises an IP protocol.

39. A computer as recited in claim 34, wherein the network communications protocol comprises UDP over IP.

40. One or more computer-readable media comprising instructions that are executable to perform actions comprising:

emulating one or more USB peripherals that respond to and create USB messages;

packaging the created USB messages in data packets formatted in accordance with a non-USB network communications protocol;

sending the data packets over network communications media using the network communications protocol.

41. One or more computer-readable media as recited in claim 40, wherein the instructions are executable to perform further actions comprising:

receiving data packets over the network communications media, wherein the data packets are formatted in accordance with a non-USB network communications protocol; and

unpackaging the USB messages from the data packets.

42. One or more computer-readable media as recited in claim 40, said wherein said emulating comprises emulating HID, bulk, and isochronous USB peripherals.

43. One or more computer-readable media as recited in claim 40, wherein the network communications protocol comprises an Ethernet communications protocol.

44. One or more computer-readable media as recited in claim 40, wherein the network communications protocol comprises an IP protocol.

45. One or more computer-readable media as recited in claim 40, wherein the network communications protocol comprises UDP over IP.